

456-TP-010-001

# **ECS Project Training Material Volume 10: Software Maintenance**

**Technical Paper**

**May 1997**

Prepared Under Contract NAS5-60000

**RESPONSIBLE ENGINEER**

---

Paul E. Van Hemel  
EOSDIS Core System Project

Date

**SUBMITTED BY**

---

Tom Hickey, M&O Deputy Manager  
EOSDIS Core System Project

Date

Hughes Information Technology Systems  
Upper Marlboro, Maryland

This page intentionally left blank.

# Abstract

---

This is Volume 10 of a series of 10 volumes containing the training material for the Pre-Release B Testbed of the Earth Observing System Data and Information System (EOSDIS) Core System (ECS). This lesson provides a detailed description of the different tasks that are required to perform software maintenance. The lesson includes a detailed review of COTS and custom software installation, troubleshooting, corrective action, and corrective action reporting.

**Keywords:** training, instructional design, course objective, software installation, software maintenance

This page intentionally left blank.

# Contents

---

## Abstract

## Introduction

Identification .....	1
Scope .....	1
Purpose .....	1
Organization .....	1

## Software Maintenance Overview

Lesson Overview .....	3
Lesson Objectives.....	3
Importance.....	5

## Software Maintenance Objectives

Maintenance Categories .....	7
------------------------------	---

## Maintenance of COTS Software

COTS Software Licenses .....	9
Installing COTS Software .....	10
Obtaining COTS Software Support.....	12
Safeguarding COTS Software Media.....	14

## **Maintenance of Custom Software**

Implementation of Modifications.....	15
Test Plans and Procedures.....	16
Installing Custom Software.....	17
Obtaining Custom Software Support .....	23
Safeguarding Custom Software Media.....	24

## **Practical Exercise**

Software Maintenance.....	25
---------------------------	----

## **Slide Presentation**

Slide Presentation Description .....	27
--------------------------------------	----

## **Figures**

1. ClearCase® View Tag Browser Screen.....	17
2. ClearCase® File Browser Screen (Main Screen).....	18

## **Tables**

1. Major COTS Software License Restrictions.....	10
--	----

# Introduction

---

## Identification

Training Material Volume 10 is part of a series of Technical Papers that will be used to teach Maintenance and Operations (M&O) concepts to the M&O staff at the following Distributed Active Archive Centers (DAACs): Langley Research Center (LaRC), National Snow and Ice Data Center (NSIDC), and EROS Data Center (EDC).

## Scope

Training Material Volume 10 describes the processes and procedures to accomplish ECS Software Maintenance functions. This lesson is designed to provide the operations staff with sufficient knowledge and information to satisfy all lesson objectives.

This document reflects the August 23, 1995 Technical Baseline maintained by the contractor Configuration Control Board (CCB) in accordance with ECS technical direction #11, dated December 6, 1994.

## Purpose

The purpose of this Technical Paper is to provide a detailed course of instruction that forms the basis for understanding ECS software maintenance. Lesson objectives are developed and will be used to guide the flow of instruction for this lesson. The lesson objectives will serve as the basis for verifying that all lesson topics are contained within this Student Guide and slide presentation material.

## Organization

This document is organized as follows:

Introduction:	The Introduction presents the document identification, scope, purpose, and organization.
Student Guide:	The Student Guide identifies the core elements of this lesson. All Lesson Objectives and associated topics are included.
Slide Presentation:	Slide Presentation is reserved for all slides used by the instructor during the presentation of this lesson.

This page intentionally left blank.

# Software Maintenance Overview

---

## Lesson Overview

This lesson will provide you with the process for maintenance of ECS COTS software and custom software, including management of software maintenance contracts and licenses, software installation, obtaining software support, and corrective actions, and corrective action reporting. It provides practical experience in using the tools you will need for software installation, resolving software problems, and minimizing system down time due to software problems.

## Lesson Objectives

**Overall Objective** - The overall objective of this lesson is proficiency in the methodology and procedures for installation, troubleshooting, and maintenance of the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) software.

**Condition** - The student will be given a workstation console with access to ECS software tools including ClearCase®, Fault/Performance Management, and Management Services Subsystem (MSS) graphical user interface (GUI) tools.

**Standard** - The student will use the tools in accordance with prescribed methods and complete required procedures without error.

**Specific Objective 1** - The student will identify and describe the process for management of software maintenance contracts with vendors.

**Condition** - The student will be given a list of types of software maintenance and maintenance personnel.

**Standard** - The student will correctly state the responsibilities of the Local Maintenance Coordinator and the ILS Logistics Engineer in COTS software maintenance and correctly describe the escalation of COTS software problems, including identification of issues related to maintenance time periods.

**Specific Objective 2** - The student will describe the management of software licenses.

**Condition** - The student will be given a list of COTS software products.

**Standard** - The student will correctly state the license restrictions on HP OpenView, AutoSys, ClearCase®, and DDTs, and correctly describe the process by which the restrictions are imposed on software access.

**Specific Objective 3** - The student will execute the procedure to install, configure, and promote COTS software using ClearCase®.

**Condition** - The student will be given a requirement to install a COTS software package, and a workstation console with access to ClearCase®.

**Standard** - The student will use ClearCase® correctly to accomplish the installation in accordance with specified procedures and without error.

**Specific Objective 4** - The student will execute the procedure for obtaining COTS software support.

**Condition** - The student will be given a workstation console with access to ECS software tools including Trouble Ticket, Fault/Performance Management, HP OpenView, and Management Services Subsystem (MSS) graphical user interface (GUI) tools.

**Standard** - The student will use the tools without error in accordance with applicable procedures to perform the required activities, including reporting of a COTS software problem, troubleshooting the problem, managing/implementing corrective action, and corrective action reporting.

**Specific Objective 5** - The student will describe the procedure for safeguarding COTS software media.

**Condition** - The student will be given a list of COTS software media.

**Standard** - The student will correctly identify the controls in place to prevent loss or unauthorized use of COTS software media.

**Specific Objective 6** - The student will execute the procedure to install, configure, and promote custom software using ClearCase®.

**Condition** - The student will be given a requirement to install a custom software package, and a workstation console with access to ClearCase®.

**Standard** - The student will use ClearCase® correctly to accomplish the installation in accordance with specified procedures and without error.

**Specific Objective 7** - The student will execute the procedure for obtaining custom software support.

**Condition** - The student will be given a workstation console with access to ECS software tools including Trouble Ticket and HP OpenView graphical user interface (GUI) tools.

**Standard** - The student will use the tools without error in accordance with applicable procedures to perform the required activities, including reporting of a custom software problem, troubleshooting the problem, managing/implementing corrective action, and corrective action reporting.

**Specific Objective 8** - The student will describe the procedure for safeguarding custom software media.

**Condition** - The student will be given a list of custom software media.

**Standard** - The student will correctly identify the controls in place to prevent loss or unauthorized use of custom software media.

## **Importance**

The Software Maintenance lesson provides students with the knowledge and skills needed for correct and efficient installation and maintenance of ECS software. It is structured to provide useful skills and knowledge concerning ECS software support processes and the tools for software installation and troubleshooting. It provides effective instruction and practical exercises for installing and maintaining ECS software in an operationally ready condition, and is therefore vital to students who are preparing for a number of different positions with responsibilities in maintaining that software readiness, including positions as:

- System Administrators, System Engineers, System Test Engineers, and Software Maintenance Engineers at the DAAC.
- System Administrators, Software Maintenance Engineers, System Engineers, and System Test Engineers at the Sustaining Engineering Organization (SEO).
- Local Maintenance Coordinator.
- ILS Logistics Engineer.

This page intentionally left blank.

# Software Maintenance Objectives

---

Software maintenance entails changes to software or related activities that are required by threatened or actual deficiencies in software performance. It is distinguished from changes that result from new or modified requirements; these changes are considered developmental changes. The objectives of software maintenance are to provide four types of maintenance, according to established priorities:

- **Adaptive Maintenance (AdM)** – this refers to maintenance performed to adapt software to changes in the data or processing environments. An example is a software upgrade performed to adapt to changes in the processing environment, such as an upgraded computer system, storage capability, or network.
- **Corrective Maintenance (CrM)** – this refers to maintenance performed to identify and correct software failures, performance failures, and implementation failures. Removal of defects in design or implementation is an example of corrective maintenance.
- **Perfective Maintenance (PfM)** – this refers to maintenance performed to enhance performance, improve cost effectiveness, improve processing efficiency, or improve maintainability. An example is the modification of a human-machine interface to improve operations or user productivity.
- **Preventive Maintenance (PvM)** – this refers to maintenance performed to prevent or reduce the impacts of anticipated problems. Examples include the reallocation of disk resources to support predicted changes in usage of short-term storage, and maintenance of libraries at a backup location.

## Maintenance Categories

Maintenance priorities are codified by ESDIS and executed in terms of procedures and policy by the Sustaining Engineering Organization (SEO), based first on ECS mission requirements and then on local DAAC/EOC/SMC operational requirements. The priorities are determined in relation to the effect of a problem on mission success, as differentiated based on scope of the impact, frequency of occurrence, and the availability of an adequate work-around. Categories of problems, as derived from NASA 420-05-03 *Performance Assurance Requirements* document, reflect the following criticality levels:

- *Category 1: System/service cannot perform critical function or imposes major safety hazard.* This refers to problems that present an immediate impact to development, operations, services, or data processing; that impose a major safety hazard to personnel, systems, or space mission resources; or that result in loss of one or more essential mission objectives.

- *Category 2: System/service substantially impaired.* This refers to problems that substantially impact development, operations, services, or data processing functions; that cause failure to operate within critical performance specifications; or that prevent effectively or efficiently fulfilling baseline requirements.
- *Category 3: System/service slightly impaired.* This refers to problems that cause degraded service but allow the mission to proceed with minor or no substantial impact to development, operations, services, or data processing functions.

All Category 1 problem reports are elevated to the Government Failure Review Board and require both Government and Contractor Project Manager approval for final close-out. The SEO also recognizes two additional categories of problem.

- *Category 4: Nuisance problem.* This refers to non-substantive problems (e.g., personal preference for arrangement of video screens, color).
- *Category 5: Closed problem.* This refers to a known issue with a prior disposition.

The analysis of maintenance priorities is based on a balanced assessment of numerous factors that affect performance parameters, including:

- Effort spent maintaining the software during each year of its production life.
- Software age (based on when it was originally developed).
- The programming language(s) in which the software was written.
- Software error information and correlation data (e.g., events, rate of occurrence, difficulty of correcting error condition).
- Software change information, indicating the number of changes requested and the difficulty of making the requested changes.
- User satisfaction survey, indicating user perception of software utility.
- Maintenance programmer's effort to fix the problem or implement a Change Control Request (CCR).
- Conformance of the software to performance and/or data models.
- Difficulties of interfacing the software with other systems because of lack of conformity.
- Effect on mission success.

Such information can serve to distinguish candidate software for replacement, maintenance action, or procedural change, or to generate success profiles for positive, useful feedback to developers. Any maintenance action must be based on consultation related to long-term planning and cost analysis impact assessment as part of the change control process.

# Maintenance of COTS Software

---

The ECS Maintenance and Operations (M&O) organization is responsible for maintenance of ECS software, as well as hardware and firmware systems delivered under the ECS contract at the ECS sites. An important element in the maintenance of commercial off-the-shelf (COTS) software items in ECS is the management of software maintenance contracts with software vendors. This element includes:

- maintaining software use licenses.
- obtaining telephone assistance in resolving COTS software problems.
- obtaining software patches.
- obtaining software upgrades.

COTS software vendor support is contracted by the ECS procurement office at the ECS Development Facility (EDF).

- First year is under warranty support.
- Subsequent support is acquired through contract extension/modification as needed.
- COTS software support contracts data are maintained in a database used by the ECS Integrated Logistic Support (ILS) office to monitor and track contract expiration dates and terms.
- Local Maintenance Coordinators (LMC) can request changes to COTS software support contracts by contacting the ILS Logistics Engineer.
  - Contact by e-mail can be made by using **ilsmaint@ecs.hitc.com** as the address.
  - Contact by telephone can be made by using the number 1-800-ECS-DATA, selecting Option #1, and then requesting the ILS Logistics Engineer.

## COTS Software Licenses

Licenses to use COTS software vary by the type of software and the software vendors' policies. License types include:

- per seat.
- per site.
- specific number of concurrent users.
- unlimited users.
- lifetime use without regard to number of users or location.

COTS software licenses are maintained in a property database. The ECS Property Administrator:

- maintains the master copy of COTS software license agreements.
- maintains the COTS software license database.
- distributes COTS software for installation at the DAACs.

The ECS Program reflects several different license restrictions based on the license types negotiated for the different COTS software products used. In general, the license restrictions are imposed through a software program that runs on a license server at the DAAC. It tracks the instances of a program in use, and when the limit is reached, it precludes access by additional operators until use falls below the limit. Table 1 lists four of the major COTS packages and identifies the license restrictions that apply to each for the Testbed.

***Table 1. Major COTS Software License Restrictions***

Software	Restriction
HP Open View	TBD
AutoSys	Only 1 instance at a time may be active
ClearCase®	5 users concurrently
DDTS	Virtually unlimited (10,000 users)

## **Installing COTS Software**

The approval of appropriate CCBs is required prior to the loading of COTS software upgrades or other packages on any ECS platform. The approval process requires systematic Configuration Management (CM) procedures and documentation to ensure appropriate control of the ECS baseline and of changes to the baseline. Once the approval is received, the site Local Maintenance Coordinator notifies those personnel who will accomplish the installation (e.g., Release Installation Team, System Administrator, Network Administrator, Software Maintenance Engineer).

For the Testbed, there is one major tool used to facilitate CM control of software installation:

- **ClearCase®** – Provides a mountable file system which is used to store version-controlled data, such as source files, binary files, object libraries and spreadsheets.

The ClearCase® tool is applied to control changes in custom software and customized portions of some COTS software packages (e.g., configuration files).

At the DAAC, the installation actions are executed by the DAAC Software Maintenance Engineer and the System Administrator. Installation proceeds systematically:

- the Version Description Document (VDD) gets final updates for system and center-specific material identified by ESDIS or the operational centers, and the final VDD is published.
- the build is installed, along with operational and user documentation.
  - ClearCase®: Multiple versions of production software always retained.
  - Scripts for System Administrator to do installation.
- controlled document updates are provided to SEO Document Maintenance and entered into the CM system.
- the CM system is updated to reflect M&O and center-specific baselines.

## Obtaining COTS Software Support

Support of COTS software involves both site capability and contracted support. Site support is provided by the System Administrator and the Software Maintenance Engineer. When site support personnel confirm that a problem is attributable to the COTS software, the COTS Software vendor's technical support center/help desk is contacted by authorized personnel at the site. When a system problem is discovered, there is an initial troubleshooting/diagnostics procedure to be followed which is generic – i.e., not limited to software problems. However, when a software problem is indicated, the procedure refers the problem to the Maintenance Coordinator for software corrective maintenance. System troubleshooting tools and principles apply:

- Software package event browser (if available) for sequence of events.
- Initial troubleshooting.
  - Review error message against software operator manual; prepare trouble ticket.
  - Review system logs for evidence of previous problems.
  - Attempt software reload.
  - If problem is software (e.g., hardware has been working and reload does not correct the problem), report it to the Maintenance Coordinator – i.e., forward the trouble ticket.

A problem that is not resolved through initial troubleshooting will often require troubleshooting teamwork by the Maintenance Coordinator, the System Administrator, and perhaps a Network Analyst. These troubleshooters may perform additional steps to resolve the problem:

- specific troubleshooting procedures described in COTS software manuals.
- review of the software vendor's web site's solutions database to learn of any solutions for similar problems.
- exercise any embedded or down-loadable software diagnostic routine that will determine the status of the COTS software on the equipment.
- exercise of any locally devised troubleshooting/diagnostic procedures.
- non-replacement intervention (e.g., adjustment of thresholds or other tunable parameters).

If the software problem is not resolved by the actions of the local staff, it may be necessary to request assistance through the Maintenance Coordinator from a maintenance contractor for on-site software support. Suppose, for example, that you are a Maintenance Coordinator and the site Software Maintenance Engineer has determined there is a problem with one of the COTS software packages used for Configuration Management, that initial troubleshooting finds the problem unable to be corrected locally, that a trouble ticket has been forwarded to you, and that

you and System Administrator are not able to resolve the problem through additional troubleshooting. The correct approach is:

- Organize the data on the problem, find data on the appropriate support provider, and update the trouble ticket with this information.
  - Locate information such as software vendor technical support center/help desk telephone numbers, names of personnel authorized (by site and software) to contact the vendor, and the authorization/access codes available to the site's Local Maintenance Coordinator from the ECS ILS office.
- Contact the support provider's technical support center/help desk to obtain on-site assistance.
  - Provide them with the background data.
  - Obtain a case reference number from them.
  - Update the trouble ticket to reflect the time and date of the call and the case number.
  - Notify the originator of the problem that the contractor has been alerted to the problem.
- Maintain coordination with the vendor for the solution and ensure compliance with Configuration Management requirements.
  - Software vendor's technical support center/help desk verifies contract support authorization and assists in pinpointing the COTS software problem to provide a recommended solution.
  - Solution may include a patch, a work-around, or a fix in a future release of the software.
  - Assist in problem resolution (If a patch exists to correct the problem, the patch will be identified and provided by the software vendor over the Internet or mailed to the requester. If a patch is required but not available, the requester and the vendor together determine the seriousness of the problem. If the problem is critical, a temporary patch or work-around may be provided, with permanent solution to be implemented in a future update or release.).
  - The DAAC and Project Configuration Control Boards (CCBs) must authorize the patch to be installed as a permanent solution. This decision may be made after the fact, in accordance with emergency procedures required to continue to operate.
- Update trouble ticket with actions taken to correct the problem and delay time experienced for the solution, including reasons for each delay.

In preparation to request software support from the software vendor to resolve the problem, use the following procedure to obtain the background information needed.

### **Obtaining On-Site Software Support: Background Information**

---

- 1** Collect information needed to obtain contract maintenance support.
  - Obtain **version, release, serial number, and location** of the failed software from the software database.
  - Obtain description of problem and symptoms from **trouble ticket**.
  - Identify the **criticality** of the COTS software experiencing the problem.
- 2** Locate information on the software support vendor.
  - Obtain **name, and telephone number** of the software support vendor.
  - Obtain **access code** needed to obtain support.
  - Obtain **telephone number** of the software support vendor's technical support center.
  - Obtain **name** of site authorized contact person.
- 3** Record data on maintenance needed and maintenance provider into the trouble ticket.

---

In unusual cases, it may be necessary to resort to non-standard software support procedures. In the event that the software support vendor's technical support center/help desk is not providing timely successful solutions, or if the maintenance action is otherwise unsatisfactory, it may be necessary to escalate the problem to the ECS Sustaining Engineering Organization (SEO). The SEO is staffed with senior systems engineers knowledgeable on COTS software and can assist with diagnosing the problem. The site Local Maintenance Coordinator may go directly to the software vendor or to the ILS Logistics Engineer to obtain an escalation of software vendor support, resulting in increased vendor management review of the problem resolution, the assignment of additional resources to resolve the problem, and/or a more highly qualified technician assigned to resolve the software problem.

### **Safeguarding COTS Software Media**

For the Testbed, COTS software media and documentation are kept in locked cabinets provided for the purpose at the DAAC. Should there be a need for access to these materials (e.g., a requirement to reload a COTS software product), keys for these cabinets are available from the Operations Coordinator during operating hours.

# Maintenance of Custom Software

---

During Testbed operations, master copies of the custom software code are maintained at the ECS Development Facility (EDF). However, there may be a need for the M&O organization to modify the configuration as established at each center.

- Software Change Manager, ClearCase®, provides the vehicle to store and maintain the local library.
- Governing policies and minimum developed software component level that may be removed from (checked out for maintenance) or reintroduced to the master library are defined by the developers' determination of code modules.
- Configuration Management (CM) requirements apply (e.g., for configuration identification, configuration change control, and configuration status accounting).

Maintenance changes to the ECS baseline may come from any of several sources, such as:

- ESDIS CCB-directed changes.
- Site-level CCB-directed changes to configuration items, subject to ESDIS delegation of responsibility for site-level control.
- Developer modifications or upgrades.
- User- or operator-initiated trouble tickets.

## Implementation of Modifications

A controlled build procedure provides structure for the implementation of changes.

- Each ECS organization selects a responsible engineer (RE) for each build.
- The SEO RE establishes the set of CCRs to be included in the system build.
- Site/center REs determine applicability of any site-unique extensions for the build.
- System and center REs establish schedules for implementation, integration, and test.
- The SEO RE maintains the integrated system- and center-specific CCR list and schedule.
- The Acceptance Test Organization provides an initial Version Description Document (VDD), which contains:
  - The CCRs incorporated into the build and their operational and/or user features.
  - The build schedule.

- ECS external interfaces affected by the build.
- ECS configuration items affected by the build.
- List of ECS documentation (e.g., design documents, procedures, help files) affected by the build.
- Test program results summary.
- Test team recommendations.
- The SEO RE maintains the VDD, updating it with authorized changes. Center REs provide appendices as needed to describe any center-unique additions/modifications to the build.
- The RE (or designated team) for a CCR uses the configuration-controlled library to obtain the correct version of the source code/files. The RE/team implements the change, performs programmer testing, and updates the documentation (design, interface, and procedures documents).

## **Test Plans and Procedures**

The test program ensures that CCRs are properly implemented and that the system is protected against introduction of defects resulting from changes.

- Program tests features (has the CCR been properly implemented) and regression (revalidation of proper operation of the configuration item and the system).
- Test team includes maintenance programmers, vendors, users, or any personnel who reported the problem that initiated the change or who use the software.
- Verification may be by inspection, analysis, or demonstration of a feature or operation, or by review of test data.
- System and center test teams develop test plans for the build, describing:
  - The CCRs to be tested.
  - The CM baseline(s) to be used.
  - The requirements and feature to be verified.
  - The method of verification, including identification of test cases/data sets.
  - Acceptance criteria.
  - Resource requirements
  - Schedule of testing.
- Feature testing is performed through development of new test cases and data or modification of existing test cases and data. Regression testing is performed using standard test cases with expected test results.

- Test teams provide test results and analyses to the SEO and center REs. A summary of the test program and the test team's recommendations are added to the VDD.

## Installing Custom Software

Just as with COTS software, the approval of appropriate CCBs is required prior to the loading of custom software upgrades or other packages on any ECS platform. The approval process, major tools used to facilitate CM control of software installation, and installation process require the same systematic steps as those associated with the installation of COTS software. However, as noted previously, the configuration of custom software is controlled by the Change Control Manager software ClearCase®.

In ClearCase®, access to versions of files in a **Versioned Object Base (VOB)** is facilitated by a **view**. When ClearCase® is initiated, the operator is asked to select a view. Available views are displayed in the View Tag Browser screen, illustrated in Figure 1.

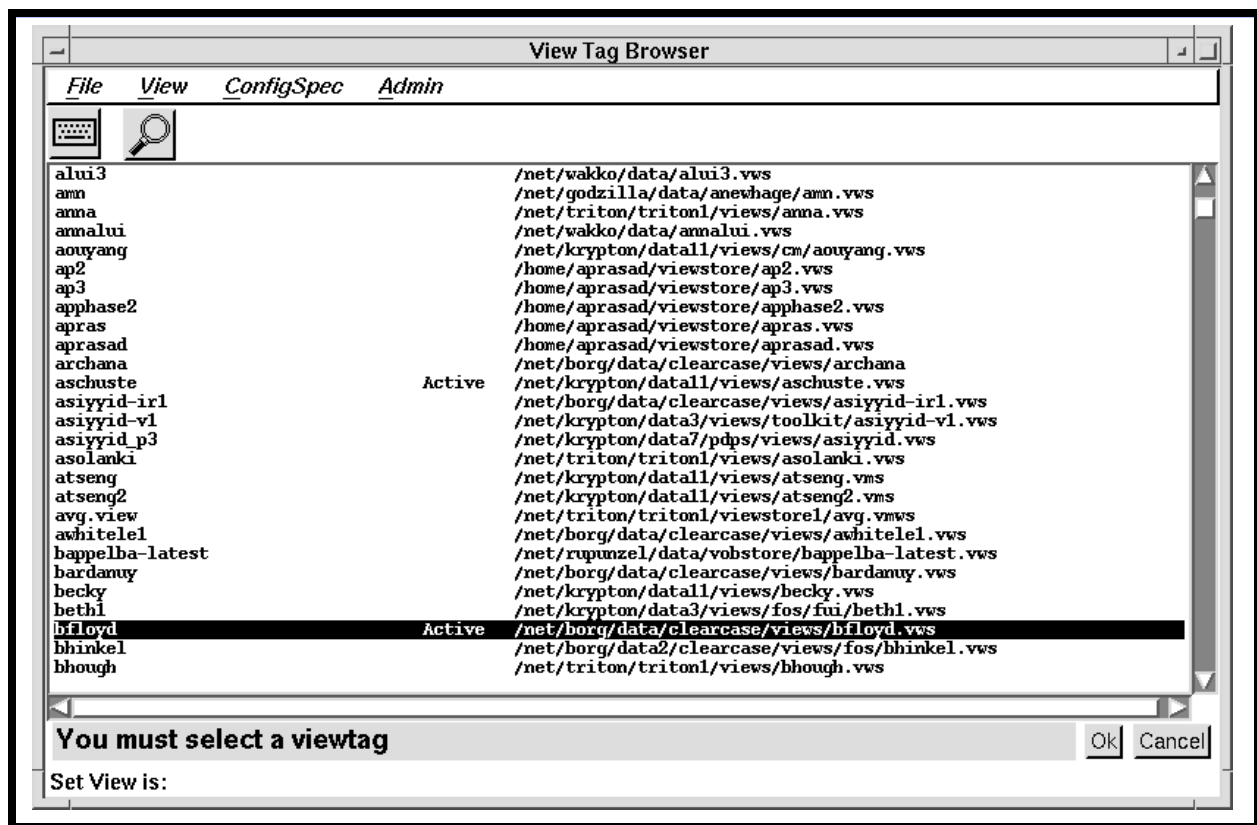


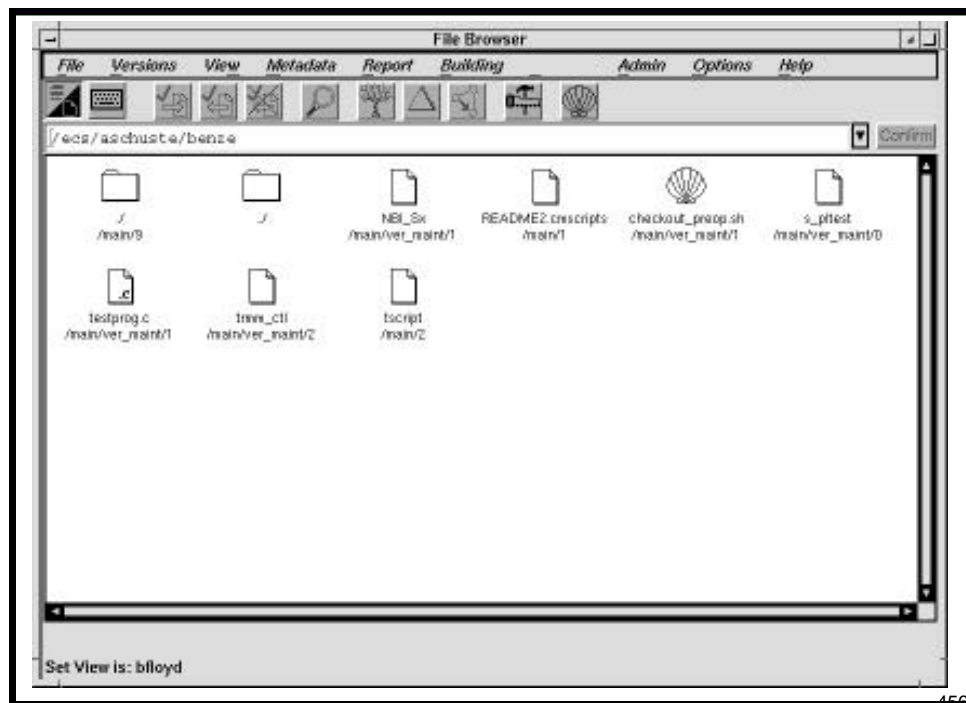
Figure 1. ClearCase® View Tag Browser Screen

Selecting a view listed in the View Tag Browser screen brings up the File Browser, or main screen, shown in Figure 2.

- Displays the directory name of the current VOB, just below the toolbar. A VOB is defined by several characteristics:
  - A mountable file system to store version-controlled data (e.g., source files, binary files, object libraries, spreadsheets).
  - Can be mounted on some or all workstations.
  - Several VOBs may exist on a machine or network.
  - When mounted as a file system of type MVFS (Multiple Version File System), a VOB can be accessed with standard UNIX and ClearCase® tools.
  - The ClearCase® file system is transparent.

A VOB consists of a storage area (for versioned files, derived objects, and cleartext files) and a database.

- Displays the content of the directory in the space below the directory's name.
- Many ClearCase® functions can be initiated from the File Browser screen.



**Figure 2. ClearCase® File Browser Screen (Main Screen)**

The steps involved in software installation at the site are documented in the following procedures.

## Software Transfer and Installation

---

- 1 Mount the software change package delivery medium (e.g., CD ROM).
- 2 On workstation **mssx2sun**, at the Unix prompt in a terminal window, to put the package on the system, type **tar -xvf /<dev/cd0** and then press the **Enter** key.
  - NOTE: The **x** in the workstation name will be a letter designating your site: **g** = GSFC, **l** = LaRC, **e** = EDC, and **n** = NSIDC (e.g., **mssn2sun** indicates a management services subsystem sun workstation at NSIDC). If you access the workstation through a remote login (rlogin), you must enter **xhost +** prior to the rlogin, and enter **setenv DISPLAY <local\_workstation IP address>:0.0** after the rlogin before entering the **tar** command.
  - The software change package directory structure is copied onto the system.
- 3 To go to the parent directory of the UNIX directory structure to be brought into ClearCase® (i.e., the software change package), type **cd parentdir** (where **parentdir** is the path of the directory that contains the directory structure to be brought into ClearCase®) and then press the **Enter** key.
  - This directory must not be in the VOB.
- 4 Create a conversion script by typing **clearcvt\_unix -r dirname**, where **dirname** is the name of the directory containing the directory structure to be brought into ClearCase® (i.e., the software change package).
  - The command “clearcvt\_unix” is a custom command that creates a script to check all the files in **dirname** into the VOB.
  - The addition of “-r” to the command ensures that any subdirectories below **dirname** will be recursively included in the script created.
  - In the **parentdir** directory, a subdirectory **cvt\_dir** is created with the script and necessary files to create the required elements within ClearCase®.
- 5 At this time the DAAC CM Administrator logs out from this workstation. The DAAC System Administrator (SA) completes the procedure.
  - The remaining steps are accomplished by the DAAC SA.

- 6 Log into the vob server platform (**mssxnsun**) by typing **username** and then pressing the **Enter** key.
  - NOTE: The **x** in the platform name will be a letter designating your site: **g** = GSFC, **l** = LaRC, **e** = EDC, and **n** = NSIDC. The **n** will be a numeral indicating the MSS workstation that is the vob server (e.g., **mssn2sun** indicates a management services subsystem sun workstation at NSIDC). If you access the workstation through a remote login (rlogin), you must enter **xhost** + prior to the rlogin.
  - The cursor moves to the **Password** field (or, if you are logging in remotely from a terminal window, a **Password:** prompt appears).
- 7 Type the **password** and then press the **Enter** key.
- 8 In a terminal window, log in as the VOB administrator by typing **su - vobadm** and then pressing the **Enter** key.
  - The prompt indicates shows that you are logged in as the **vobadm**.
- 9 Type **setenv DISPLAY <local\_workstation IP address>:0.0** and then press the **Enter** key.
- 10 To change to the ClearCase® start-up directory, type **cd /usr/atria/bin** and then press the **Enter** key.
  - The prompt shows the directory as **/usr/atria/bin**.
- 11 Type **xclearcase &** and then press the **Enter** key.
  - The ClearCase® **Transcript** screen is displayed as the View Tag Browser loads.
  - The ClearCase® **View Tag Browser** screen is displayed listing available views.
- 12 Click on the desired view name to highlight (select) it and display its name in the **Select view tag** field, and then press the **Enter** key or click on the **OK** button.
  - The ClearCase® **Transcript** screen is displayed as the File Browser screen loads.
  - The ClearCase® **File Browser** screen is displayed.
- 13 To create a view for checking in the software change package, follow menu path **File→Execute→Single Command**.
  - A String Browser window is displayed containing a field labeled **Enter shell command to run:**.
- 14 Type **"mkview viewname"** and press the **Enter** key or click on the **OK** button, where **viewname** is the name of the new view (e.g., **SAName\_changepackage**).
  - A terminal window displays a message that the create view is successfully executed.

- 15 Follow menu path **View**→**Set** to return to the **View Tag Browser** to set the newly created view for the software change package.
- 16 Click on the name of the view for the software change package to highlight it, and then click on the **OK** button at the bottom of the screen..
  - The ClearCase® **File Browser** screen is displayed with the new view set.
- 17 Type into the directory input box of the **File Browser** the name of the directory in the VOB (e.g., **/vobhome/SSIT/**) where the software change package is to be imported and then press the **Enter** key or click on the **Confirm** button.
  - The display shows the files in the entered directory.
- 18 To create the new directory under ClearCase®, follow menu path **File**→**Execute**→**Single Command**.
  - A String Browser window is displayed containing a field labeled **Enter shell command to run:**.
- 19 Type **/<parentdir>/cvt\_dir/cvt\_script** and press the **Enter** key or click on the **OK** button, where **parentdir** specifies the path/location of the **cvt\_dir** created in **Step 4**.
  - A terminal display window appears and displays a message **Converting files . . .** and confirming other script actions (e.g., checkout of the directory, creation of ClearCase® elements, checkin of the directory, and setting of permissions).
- 20 On the **File Browser** screen, follow menu path **File**→**Exit**.
  - The ClearCase® Graphical User Interface session is closed.

---

The Testbed does not include a Baseline Manager software tool (XRP-II is to be provided as part of later releases). Nevertheless, it is recommended that you keep a written record to document the changes. As a minimum, it is desirable to record:

- DAAC CCR Number.
- Software Package Identification.
- Package Name.
- Software Upgrade Name.
- Version.
- File Structure.
- Type.
- Installation Date.

The DAAC Software Maintenance Engineer implements and tests the new software:

- Initiates software transfer to compiler hosts for make and build.
- Initiates software distribution.
- Tests individual packages (unit, subsystem, system).
- Runs the full final software in the operational environment.
- Notifies the SEO of the results.

Scheduled maintenance is a preferred approach for controlling the maintenance function applied to typical custom application programs. Emergency fixes and work-arounds are applied as required, but all other repairs or changes are assessed for the determination of an appropriate maintenance schedule. This approach offers several benefits:

- By consolidating the changes to be made to a configuration item, modifications can be performed more efficiently (e.g., with reduced numbers of documentation updates, minimized disruptions to operations, and decreased costs).
- Because requesters know that their changes will not be acted upon immediately, they can give more consideration to what changes are actually needed.
- Aggregating changes in batches permits efficiency and integration in assessment, with more thorough evaluation.
- Insight into the integration of maintenance into monthly/yearly cycles facilitates management by allowing more effective setting of priorities for maintenance projects.
- Positive control of baseline management across M&O and Development organizations.

The VDD is the vehicle for communicating the contents, status, features, schedule, and test results to ECS operations and users.

- Test plans, test procedures, and test results supplement the VDD.
- Draft and final versions of the VDD are published and distributed to interested parties:
  - Internal organizations (e.g., ECS Development Offices, System Management Office, Quality Office, Science Office).
  - External organizations (e.g., ESDIS, DAAC, other customer, external systems, Independent Verification and Validation contractor, Science Computing Facilities, user groups).

The golden copy of ECS custom software is maintained by the SEO CM Administrator. Required access to the golden copy, as well as changes, is guaranteed by logging changes and backup of modifications for later access as required by users, developers, and maintenance

personnel. Custom software is also maintained by the CM Administrator at the ECS deployment sites.

## **Obtaining Custom Software Support**

A list of Responsible Engineers for the software configuration items provides current points of contact for help. On-site Maintenance Engineers can use this list and consult with SEO experts who can lead troubleshooting activities for specific configuration items. The activity can be coordinated through use of trouble ticket priorities, including provision of emergency fixes. Permanent changes are arranged through use of CCRs.

The maintenance of science software and data items provided by the Science Computing Facilities (SCFs) is not the responsibility of the ECS on-site maintenance engineers. Problem resolutions and changes to this software will be handled under the auspices of local DAAC CM activities and the ESDIS CCB in the same manner as new releases to baselined science software.

Problems with ECS custom software are one type of impetus for generating trouble tickets (TTs):

- Anomalies.
- Apparent incorrect execution by an ECS software configuration item.
- Inefficiencies.
- Sub-optimal use of system resources.
- TTs may be submitted by users, operators, customers, analysts, maintenance personnel, and management staff.
- TTs capture supporting information and data related to the problem.

Troubleshooting is conducted on an ad hoc basis. Just as with COTS software problems, however, it is conducted systematically.

- Site-level activity is initiated by the Operations Supervisor assigning a trouble ticket to the Problem Investigator.
- Problem Investigator uses list of Responsible Engineers if needed to obtain support from SEO Maintenance Programmers, Responsible Engineers, and ECS Developers at the ECS Development Facility (EDF).
- EDF has the same software and computer equipment variants available at the sites, and may be able to duplicate anomalies experienced in an on-site system to derive effective resolutions or work-arounds as required until a permanent solution can be implemented.
- At a TT telecon, the Failure Review Board assigns a priority to the TT and assigns the TT to an organization for work-off. The organization assigns a Responsible Engineer to work off the TT. Using the data captured in the TT, the Responsible Engineer conducts a technical investigation to attempt to isolate the source of the reported problem.

- If the problem is caused by a non-ECS element (e.g., an interface problem with an external system, poor resource usage by a science algorithm, poor performance by a non-ECS service), the TT and supporting data are provided to the maintainer of that element. An ECS CCR may also be proposed to protect ECS from potential threats of future problems identical or similar to that documented in the TT.
- If a TT is properly written against an ECS element, one or more of the following actions result:
  - Describe the sources of the problem and the recommended design/implementation change. Procedure modifications may also be appropriate.
  - Modify procedures. Describe the source of the problem and modify procedures to eliminate or reduce the number of occurrences of the documented problem. Modifications may be temporary (i.e., work-around) or permanent. If the change is permanent, the TT can be closed and/or a User Recommendations Data Base (URDB) input generated.
  - Track. The technical investigation focuses on collection of additional data from new occurrences to support additional analyses into the root of the problem and/or the frequency of occurrence. As a result of tracking, further technical investigations may result in any of the other actions.
  - Reset priorities. Describe the results of the technical investigation and recommend a priority change at the TT telecon. A lowered priority may result in the TT going into backlog status or being closed. A higher priority may result in additional resources being applied to the technical investigation.
  - Close with URDB input. The technical investigation may discover that what is being reported as a problem is actually the proper implementation of the feature based on the requirements baseline. A URDB input documents a recommended requirements change.
  - Close TT into existing TT or CCR. If the TT documents a known problem for which no solution has been identified, the new TT can be closed into the existing TT. Supporting material from the new TT is added to that previously collected. The TT may also be closed into a CCR that has been previously written but not yet installed into the operational baseline.

## **Safeguarding Custom Software Media**

As noted, the master copies of custom software code for the Testbed are maintained at the ECS Development Facility (EDF). If it is necessary to reload custom software, this can be accomplished through telnet.

# Practical Exercise

---

## Introduction

This exercise is designed to practice key elements of typical software maintenance operations procedures. Perform the tasks identified in the exercises.

## Equipment and Materials

ECS workstations.

## Software Maintenance

1. List the telephone numbers for the technical support center/help desks and names of personnel authorized to contact the vendor for the following COTS software products:
  - HP OpenView.
  - AutoSys.
  - ClearCase®.
  - DDTs.
2. You are provided with a custom software change package. Check the files into ClearCase®.

This page intentionally left blank.

# Slide Presentation

---

## Slide Presentation Description

The following slide presentation represents the slides used by the instructor during the conduct of this lesson.